# Proposed Photino Native Menu API

Mike Yeager

Thursday, April 1, 2021

## Types of Set menu

There are 4 types of native menus: System menu, App menu, Floating Popup (context) menus and Dock/Taskbar menus. We are initially concerned with only the native App menu.

## App Menu Rules

Should we allow 1 App menu per window in a Photino application? Or 1 App menu per application on the “main” window? Should we use ints or guids for Ids?

START WITH THE 1 MENU PER WINDOW APPROACH

## Proposed App Menu API

int Id = **InsertMenuItem**(

int? parentMenuId,

int position,

string text,

bool isEnabled = true,

bool isChecked = false);

int Id = **InsertMenuSeparator**(int parentMenuId, int position);

bool Success = **DeleteMenuItem**(int id);

bool IsEnabled = **GetMenuItemIsEnabled**(int id);

bool Success = **SetMenuItemIsEnabled**(int id, bool isEnabled);

bool IsChecked **GetMenuItemIsChecked**(int id);

bool Success = **SetMenuItemIsChecked**(int id, bool isChecked);

bool Success = **SetMenuItemText**(int id, string text);

int Count = **GetMenuItemCount**(int id);

bool Success = **DeleteMenu**(int? id);

bool Success = **LoadMenu**(string jsonMenuDefinition);

## Proposed JSON Menu Definition

Allows users to pre-define an entire menu and load it instead of building a menu in code. This menu can be altered using the above methods.

Example:

{

“Items”: [

{

“Id”: 100,

“Text”: “&File”,

“Items”: [

{

“Id”: 101,

“Text”: “&New”

},

{

“Id”: 102,

“Text”: “&Open”,

“IsEnabled”: false

},

{

“Id”: 103,

“Text”: “&Save”

},

{

“Id”: 104,

“Text”: “Save &As…”

},

{

“Id”: 105,

“Text”: “Aut&o-Save”,

“IsChecked”: true

},

{

“Id”: 110,

“IsSeparator”: true

},

{

“Id”: 115,

“Text”: “E&xit”

}

]

},

{

“Id”: 200,

“Text”: “&Edit”,

“Items”: [

{

“Id”: 201,

“Text”: “&Undo”

},

{

“Id”: 210,

“IsSeparator”: true

},

{

“Id”: 211,

“Text”: “C&ut”

},

{

“Id”: 212,

“Text”: “&Copy”

},

{

“Id”: 213,

“Text”: “&Paste”

}

]

},

{

“Id”: 900,

“Text”: ”&Help”

“Items”: [

{

“Id”: 901,

“Text”: “&About”

}

]

}

]

}

## Potential Future Enhancements

Multiple columns, icons

## Events

When the above methods are called and the call is successful, a corresponding event is raised passing all of the parameters to the call to all registered listeners:

**OnInsertMenuItem**

**OnInsertMenuSeparator**

**OnDeleteMenuItem**

**OnSetMenuItemIsEnabled**

**OnSetMenuItemIsChecked**

**OnSetMenuItemText**

**OnDeleteMenu**

**OnLoadMenu**

When a selection is made from the menu, a corresponding even is raised passing the following parameters to the call to all registered listeners:

**OnMenuSelection**(int id, string text, bool isChecked)